

Rendu TP Méthodes Numériques :

Simulation d'écoulement fluide

Jacopo Iollo, Geoffroy Oudoumanessah

1 Résolution de l'équation de transport diffusion

Question 1 : On cherche deux matrices M et N telles que $\forall k \in 0, \dots, N_t - 1$ on ait le système :

$$N \begin{pmatrix} \Phi(0, (k+1)dt) \\ \vdots \\ \Phi(ndx, (k+1)dt) \\ \vdots \\ \Phi((N_x - 1)dx, (k+1)dt) \end{pmatrix} = M \begin{pmatrix} \Phi(0, kdt) \\ \vdots \\ \Phi(ndx, kdt) \\ \vdots \\ \Phi((N_x - 1)dx, kdt) \end{pmatrix}$$

Pour répondre à cette question on utilise la formule de l'équation différentielle discrétisée et on sépare les termes en $\Phi(., t + dt)$ et $\Phi(., t)$, on obtient alors :

$$\begin{aligned} & \Phi(x, t + dt) - \kappa \frac{dt}{dx^2} (\Phi(x + dx, t + dt) - 2\Phi(x, t + dt) + \Phi(x - dx, t + dt)) \\ &= \Phi(x, t) - c(x) \frac{dt}{2dx} (\Phi(x + dx, t) - \Phi(x - dx, t)) + c(x)^2 \frac{dt^2}{2dx^2} (\Phi(x + dx, t) - 2\Phi(x, t) + \Phi(x - dx, t)) \end{aligned}$$

Puis finalement, en arrangeant les termes on trouve la forme suivante :

$$\begin{aligned} & -\kappa \frac{dt}{dx^2} \Phi(x - dx, t + dt) + (1 + 2\kappa \frac{dt}{dx^2}) \Phi(x, t + dt) - \kappa \frac{dt}{dx^2} \Phi(x + dx, t + dt) \\ &= (c(x) \frac{dt}{2dx} + c(x)^2 \frac{dt^2}{2dx^2}) \Phi(x - dx, t) + (1 - 2c(x)^2 \frac{dt^2}{2dx^2}) \Phi(x, t) + (-c(x) \frac{dt}{2dx} + c(x)^2 \frac{dt^2}{2dx^2}) \Phi(x + dx, t) \end{aligned}$$

Enfin, en utilisant la périodicité spatiale de Φ on en déduit aisément les matrices :

$$N = \begin{pmatrix} 1 + 2\kappa \frac{dt}{dx^2} & -\kappa \frac{dt}{dx^2} & 0 & \dots & -\kappa \frac{dt}{dx^2} \\ -\kappa \frac{dt}{dx^2} & 1 + 2\kappa \frac{dt}{dx^2} & -\kappa \frac{dt}{dx^2} & \dots & \vdots \\ 0 & -\kappa \frac{dt}{dx^2} & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\kappa \frac{dt}{dx^2} \\ -\kappa \frac{dt}{dx^2} & \dots & 0 & -\kappa \frac{dt}{dx^2} & 1 + 2\kappa \frac{dt}{dx^2} \end{pmatrix}$$

$$M = \begin{pmatrix} a_1 & b_1 & 0 & \dots & c_1 \\ c_2 & a_2 & b_2 & \dots & \vdots \\ 0 & c_3 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & b_{N_x-1} \\ b_{N_x} & \dots & 0 & c_{N_x} & a_{N_x} \end{pmatrix}$$

avec $\forall k \in 0, \dots, N_x - 1$

$$\begin{cases} a_k = 1 - 2c(kdx)^2 \frac{dt^2}{2dx^2} \\ b_k = c(kdx) \frac{dt}{2dx} (c(kdx) \frac{dt}{dx} - 1) \\ c_k = c(kdx) \frac{dt}{2dx} (c(kdx) \frac{dt}{dx} + 1) \end{cases}$$

Question 2 : On a bien N qui est symétrique, pour vérifier qu'elle est définie positive,

on va poser la matrice de Jordan : $J = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \end{pmatrix}$. On montre que les valeurs

propres de la matrice de Jordan sont les racines N_t^{ime} de l'unité. On trouve par ailleurs que N est la matrice issue de l'évaluation d'un certain polynôme par J . En posant ce polynôme Q , on trouve alors :

$$Q(X) = 1 + 2\kappa \frac{dt}{dx^2} - \kappa \frac{dt}{dx^2} (X + X^{N_t-1})$$

On a aussi J diagonalisable $\exists P \in GL_n(\mathbb{C})$ telle que $J = P^{-1}DP$ avec $D = \text{Diag}((\omega_k)_{0 \leq k \leq N_t})$, ainsi on obtient alors $Q(J) = P^{-1}Q(D)P$ par bloc on obtient que les valeurs propres de la matrice N sont les évaluations de Q en $(\omega_k)_{0 \leq k \leq N_t}$. Montrons que $\forall k \in 0, \dots, N_t-1$ on a

$Q(\omega_k) > 0$. En effet on a bien $\omega_k + \omega_k^{N_t-1} = e^{\frac{2ik\pi}{N_t}} + e^{\frac{2ik\pi(N_t-1)}{N_t}} = 2\cos(\frac{2k\pi}{N_t}) \in \mathbb{R}$ de plus $1 + 2\kappa \frac{dt}{dx^2} - \kappa \frac{dt}{dx^2} (2\cos(\frac{2k\pi}{N_t})) > 0$, N est alors bien symétrique définie positive.

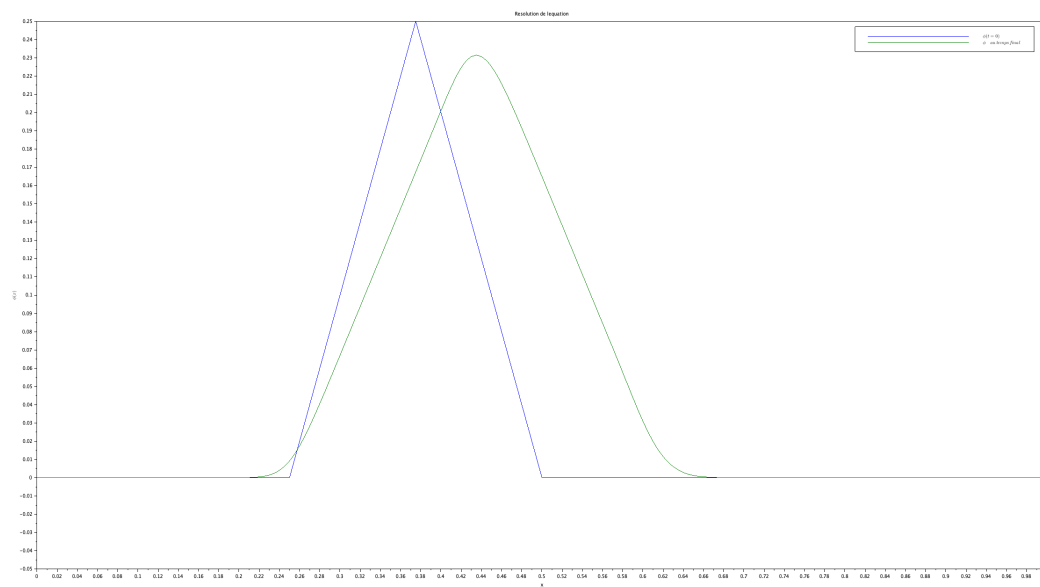
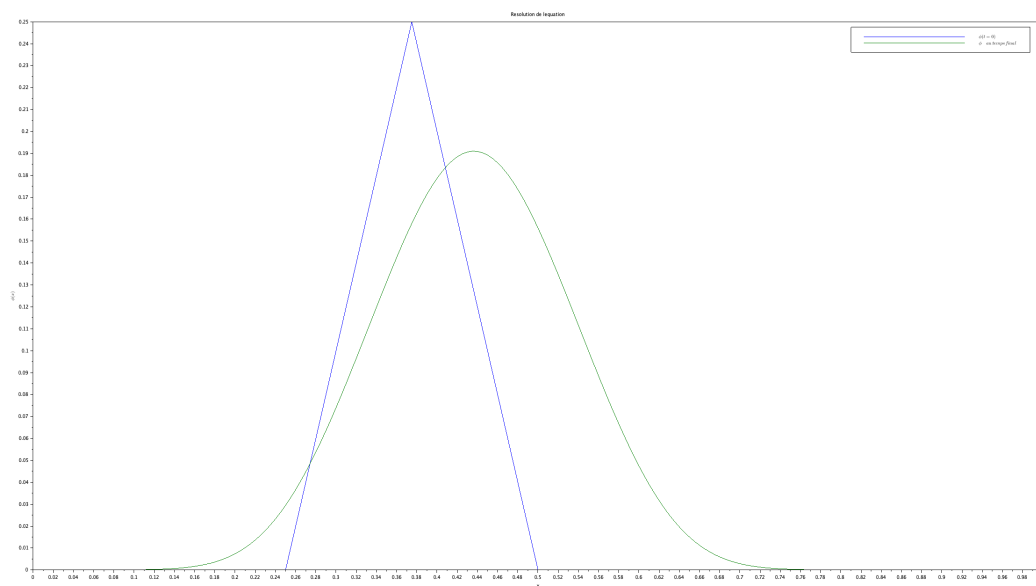
Cela justifie donc bien l'utilisation d'une méthode de type Cholesky car on peut écrire, avec $T \in T_n(\mathbb{R})$:

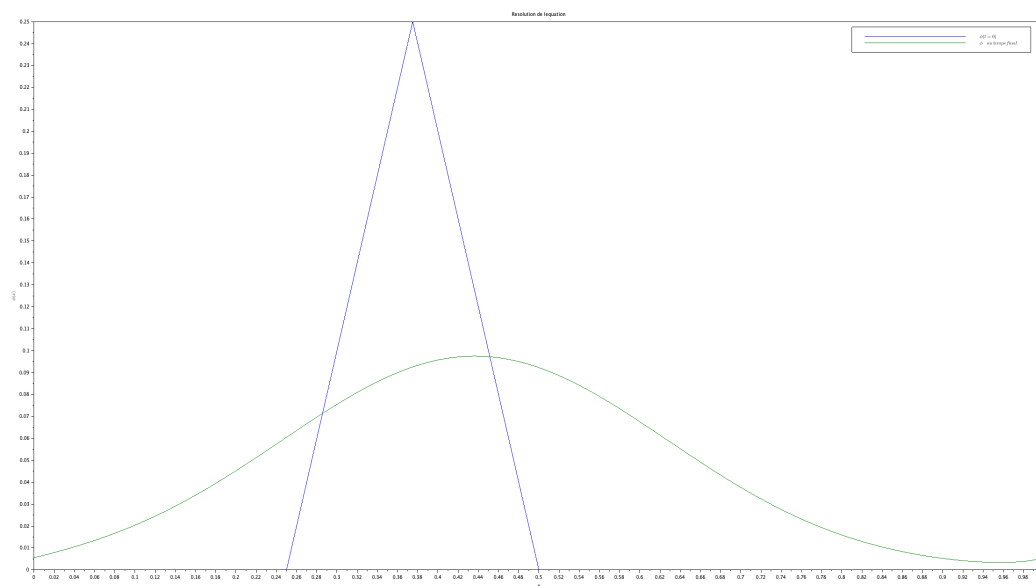
$$T^t T \begin{pmatrix} \Phi(0, (k+1)dt) \\ \vdots \\ \Phi(ndx, (k+1)dt) \\ \vdots \\ \Phi((N_x-1)dx, (k+1)dt) \end{pmatrix} = M \begin{pmatrix} \Phi(0, kdt) \\ \vdots \\ \Phi(ndx, kdt) \\ \vdots \\ \Phi((N_x-1)dw, kdt) \end{pmatrix}$$

$$\text{On résout alors : } T^t Y = M \begin{pmatrix} \Phi(0, kdt) \\ \vdots \\ \Phi(ndx, kdt) \\ \vdots \\ \Phi((N_x-1)dw, kdt) \end{pmatrix} \text{ et } T \begin{pmatrix} \Phi(0, (k+1)dt) \\ \vdots \\ \Phi(ndx, (k+1)dt) \\ \vdots \\ \Phi((N_x-1)dx, (k+1)dt) \end{pmatrix} = Y$$

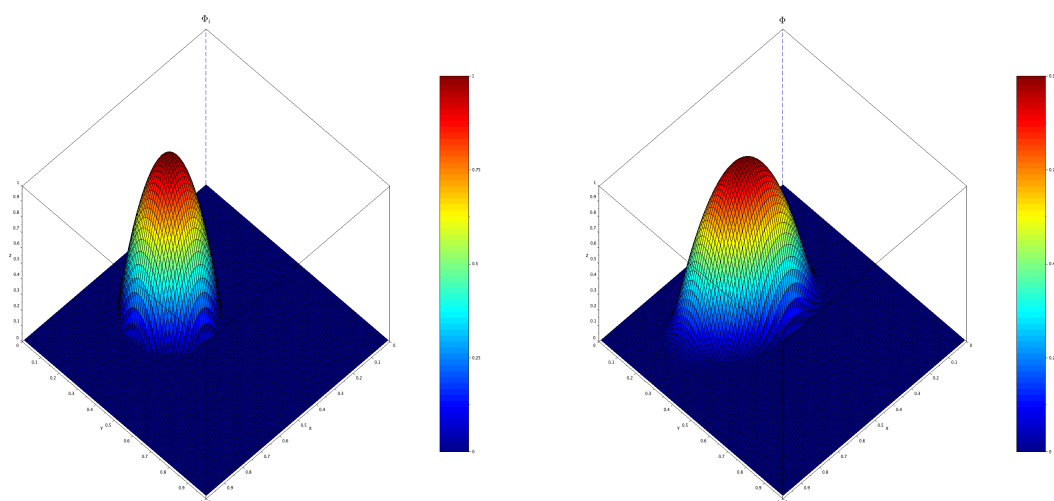
Question 3 : On code les fonctions `up_sweep_cholesky` et `down_sweep_cholesky` avec l'algorithme de Cholesky vu en cours. Enfin on utilise l'algorithme de résolution vu précédemment pour coder la fonction `my_cholesky`.

Question 4 : On initialise d'abord les matrices M , N , ϕ_0 et la fonction de convection. Enfin on applique N_t fois la fonction `my_cholesky` pour trouver Φ au temps final on obtient, pour els différentes valeur de κ testées :

FIGURE 1 – $\kappa = 10^{-4}$ FIGURE 2 – $\kappa = 10^{-3}$

FIGURE 3 – $\kappa = 10^{-2}$

Question 5 : On code assez simplement la fonction `solveur_1D` avec la fonction `umfpack` qui résous un système de la forme $Ax = b$. Enfin on écrit une fonction `solveur_2D` qui utilise la méthode de splitting présentée et on plot les résultats. On obtient alors :

FIGURE 4 – *Solveur2D*

2 Résolution du problème de Poisson

Question 6 : On a, $\forall n, m > 1$ et pour $\mathbf{k}_x = \frac{2i\pi q}{L_x}$ et $\mathbf{k}_y = \frac{2i\pi p}{L_y}$: $S_{nm}(f) = \sum_{p=-n}^n \sum_{q=-m}^m \hat{f}_{pq} e^{\mathbf{k}_y y} e^{\mathbf{k}_x x}$ et $S_{nm}(\Psi) = \sum_{p=-n}^n \sum_{q=-m}^m \hat{\Psi}_{pq} e^{\mathbf{k}_y y} e^{\mathbf{k}_x x}$. De plus, on peut écrire :

$$\frac{\partial^2 S_{nm}(\Psi)}{\partial x^2}(x, y) + \frac{\partial^2 S_{nm}(\Psi)}{\partial y^2}(x, y) = S_{nm}(f)(x, y)$$

D'où, en passant faisant la somme des dérivées :

$$\sum_{p=-n}^n \sum_{q=-m}^m \hat{\Psi}_{pq} \mathbf{k}_x^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} + \sum_{p=-n}^n \sum_{q=-m}^m \hat{\Psi}_{pq} \mathbf{k}_y^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} = \sum_{p=-n}^n \sum_{q=-m}^m \hat{f}_{pq} e^{\mathbf{k}_y y} e^{\mathbf{k}_x x}$$

Puis finalement, en identifiant les coefficients, on trouve alors :

$$\hat{\Psi}_{pq} = \frac{\hat{f}_{pq}}{\mathbf{k}_x^2 + \mathbf{k}_y^2}$$

Question 7 : Pour cette question, on écrit la fonction `fftfreq` avec la formule donnée.

Question 8 : Il suffit de coder la fonction `poisson_2d` en suivant l'algorithme donnée et en faisant attention au cas où \mathbf{k}_x et \mathbf{k}_y sont simultanément nuls.

Question 9 : Soit $\alpha \in \mathbb{R}$ et on pose $\Psi_\alpha(x, y) = \alpha \sin(2\pi x) \sin(2\pi y)$, on a :

$$\Delta \Psi_\alpha(x, y) = -8\pi^2 \alpha \sin(2\pi x) \sin(2\pi y) = \sin(2\pi x) \sin(2\pi y)$$

D'où $\alpha = \frac{-1}{8\pi^2}$ puis $\Psi_\alpha(x, y) = \frac{-1}{8\pi^2} \sin(2\pi x) \sin(2\pi y)$, on peut finalement coder la fonction `solution_field`.

Question 10 : On obtient un test réussi avec une erreur de : $8.6736173799e^{-18}$, on a donc une précision de l'ordre de 10^{-18} sur nos calculs qui sont alors très précis.

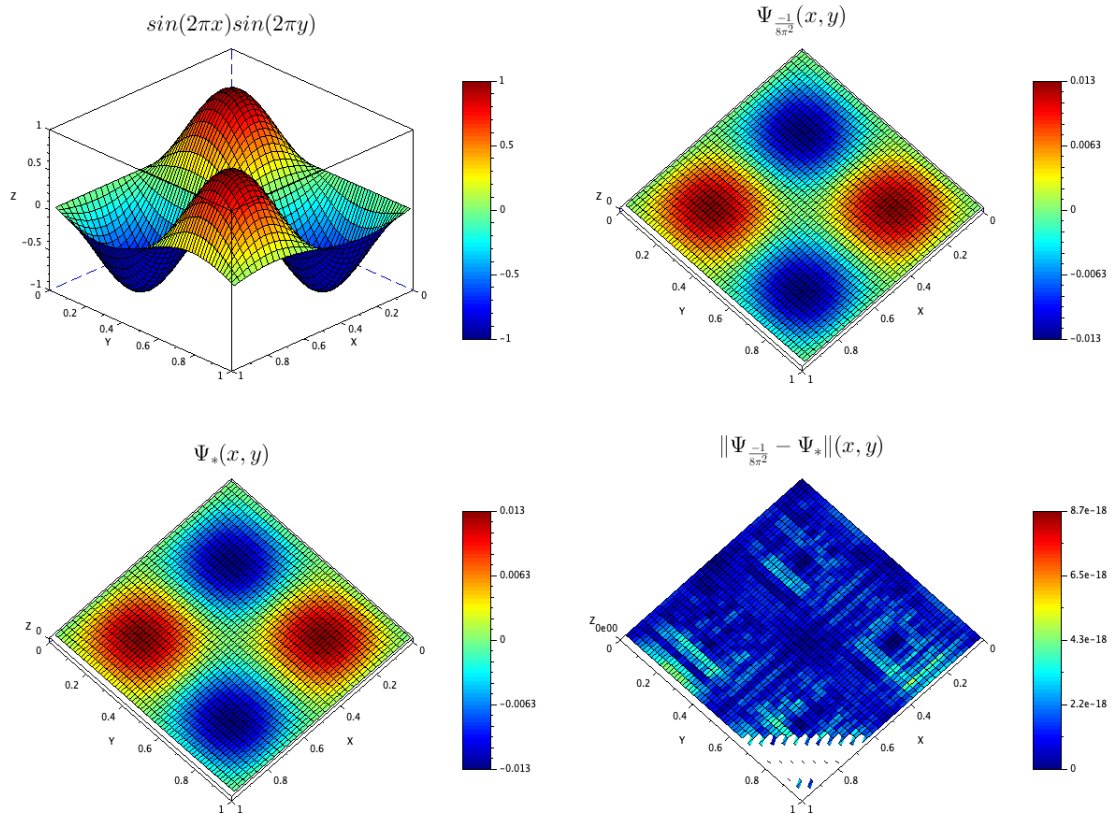


FIGURE 5 – Résultats du Test

Question 11 : On trouve les coefficients de Fourier de $\hat{\mathbf{u}}_x$ et $\hat{\mathbf{u}}_y$ avec le même raisonnement que lors de la question 8 :

$$\Delta S_{nm}(\mathbf{u}_x)(x, y) = -\frac{\partial S_{nm}(\omega)}{\partial y}(x, y)$$

$$\sum_{p=-n}^n \sum_{q=-m}^m \hat{\mathbf{u}}_{xpq} \mathbf{k}_x^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} + \sum_{p=-n}^n \sum_{q=-m}^m \hat{\mathbf{u}}_{ypq} \mathbf{k}_y^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} = -\sum_{p=-n}^n \sum_{q=-m}^m \hat{\omega}_{pq} \mathbf{k}_y e^{\mathbf{k}_y y} e^{\mathbf{k}_x x}$$

$$\Delta S_{nm}(\mathbf{u}_y)(x, y) = \frac{\partial S_{nm}(\omega)}{\partial x}(x, y)$$

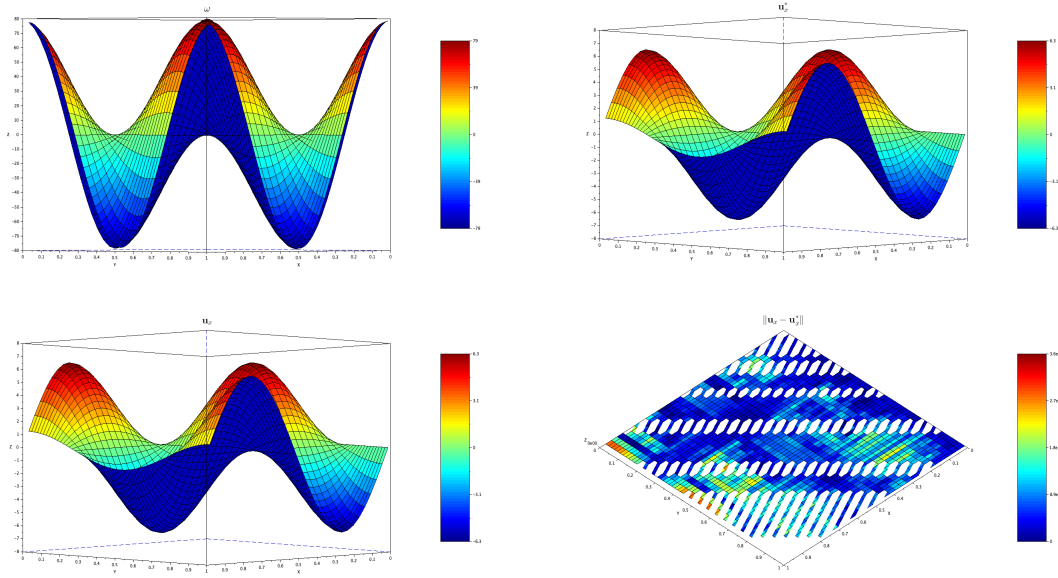
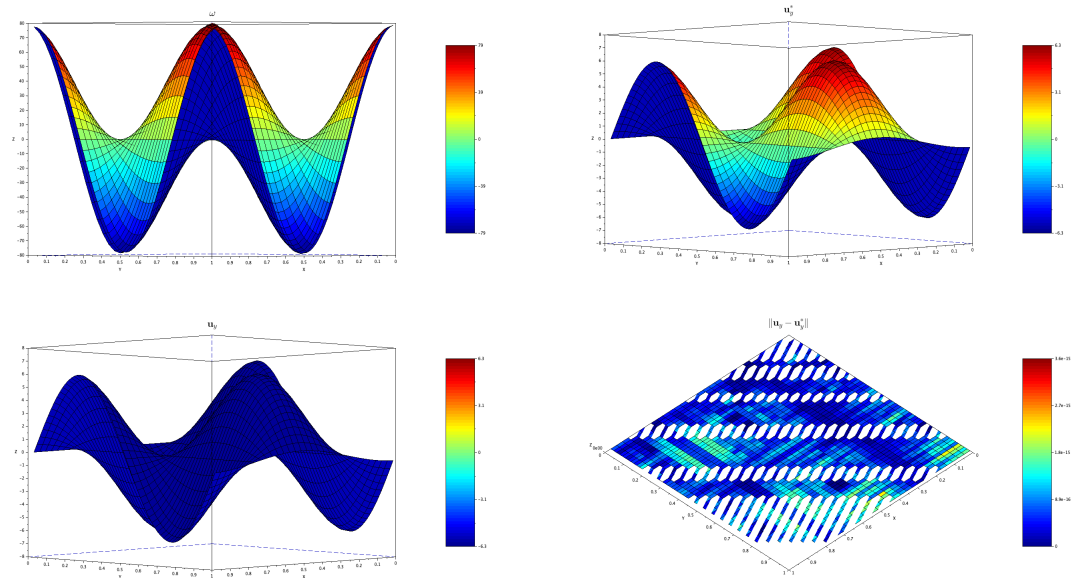
$$\sum_{p=-n}^n \sum_{q=-m}^m \hat{\mathbf{u}}_{xpq} \mathbf{k}_x^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} + \sum_{p=-n}^n \sum_{q=-m}^m \hat{\mathbf{u}}_{ypq} \mathbf{k}_y^2 e^{\mathbf{k}_y y} e^{\mathbf{k}_x x} = \sum_{p=-n}^n \sum_{q=-m}^m \hat{\omega}_{pq} \mathbf{k}_x e^{\mathbf{k}_y y} e^{\mathbf{k}_x x}$$

Finalement on a les relations suivantes :

$$\hat{\mathbf{u}}_{xpq} = \frac{-\hat{\omega}_{pq}}{\mathbf{k}_x^2 + \mathbf{k}_y^2}$$

$$\hat{\mathbf{u}}_{ypq} = \frac{\hat{\omega}_{pq}}{\mathbf{k}_x^2 + \mathbf{k}_y^2}$$

On code ainsi, dans la fonction `poisson_curl_2d` en calculant les $\hat{\mathbf{u}}_{xpq}$ et $\hat{\mathbf{u}}_{ypq}$ puis en passant à la transformée inverse de Fourier on trouve \mathbf{u}_x et \mathbf{u}_y . Pour le test donné, on obtient les résultats suivants :

FIGURE 6 – Résultats du Test pour u_x FIGURE 7 – Résultats du Test pour u_y

On constate que les erreurs commises sont de l'ordre que celles commises précédemment lors de la question 10, ce qui semble normal puisque l'on utilise la même méthode pour la résolution.

3 Simulation numérique

Question 12 : On a $\omega^0(x, y) = \frac{\partial \mathbf{u}_y^0}{\partial x} - \frac{\partial \mathbf{u}_x^0}{\partial y}$. De plus, $\frac{\partial \mathbf{u}_y^0}{\partial x} = 2\pi\delta\cos(2\pi x)$ et

$$\frac{\partial \mathbf{u}_x^0}{\partial y} = \begin{cases} \frac{\rho}{ch^2(\rho(y-0.25))} & \text{si } \rho \leq 0.5 \\ \frac{-\rho}{ch^2(\rho(0.75-y))} & \text{si } \rho > 0.5 \end{cases}$$

Question 13 :

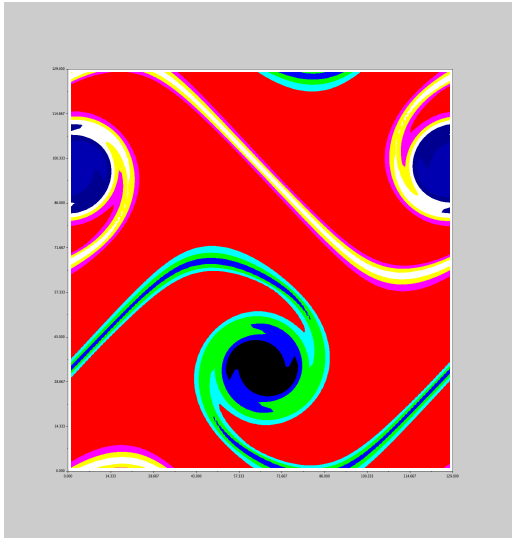


FIGURE 8 – Isocontours à $t = 1.2$

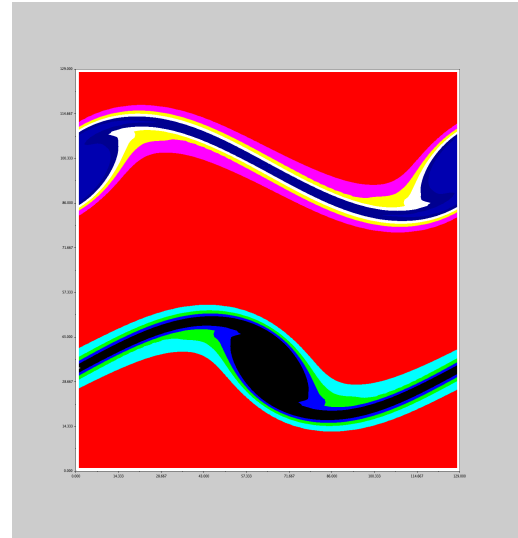


FIGURE 9 – Isocontours à $t = 0.8$

Pour $t = 0.12$ les lignes de niveau sont plus resserrées, ce qui signifie que le champ scalaire de la vorticité varie rapidement lorsqu'on se déplace orthogonalement aux isocontours. Pour $t = 0.8$, on a des isocontours plus espacés, la vorticité varie moins rapidement. Physiquement, on assiste à la "création" du tourbillon.

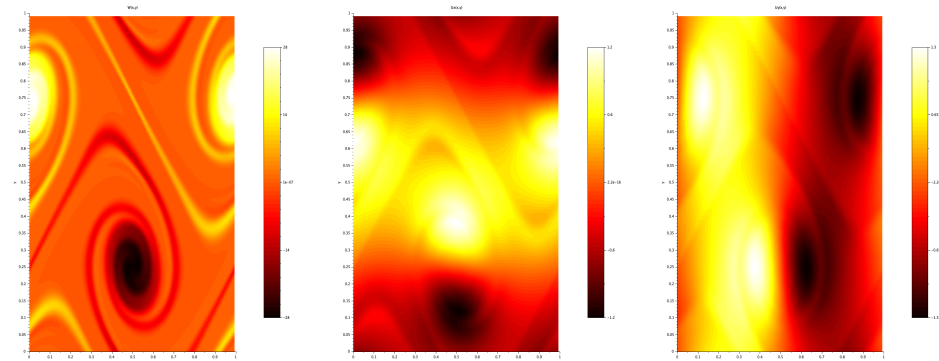
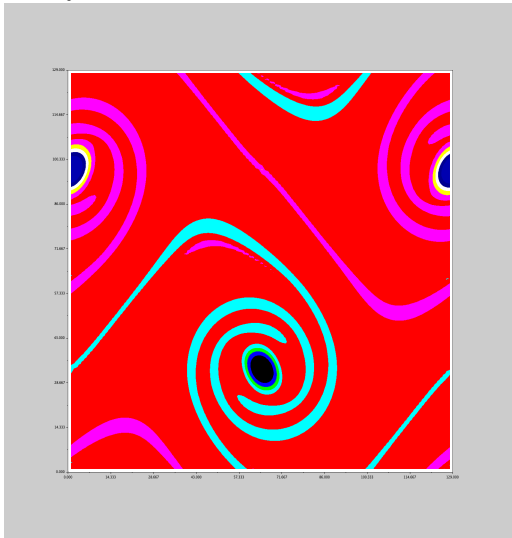
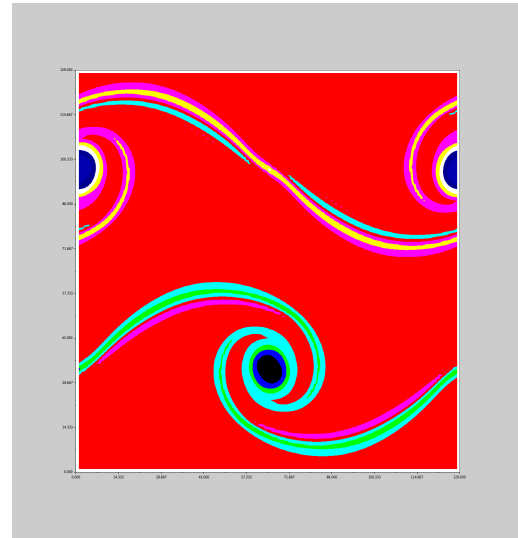


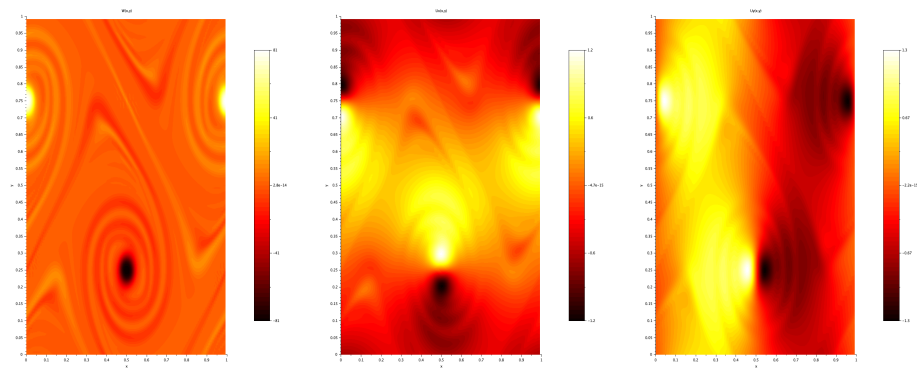
FIGURE 10 – ω , $u_x(x, y)$, $u_y(x, y)$ pour $\nu = 10^{-4}$ et $\rho = 30$

On a 240 itérations pour cette simulation.

Question 14 :

FIGURE 11 – Isocontours à $t = 1.2$ FIGURE 12 – Isocontours à $t = 0.8$

Pour ces paramètres, on a un tourbillon plus important que celui de la question précédente. On a du itérer 294 fois, ce qui rend cette configuration plus difficile à simuler que la première.

FIGURE 13 – ω , $u_x(x, y)$, $u_y(x, y)$ pour $\nu = 5 \cdot 10^{-5}$ et $\rho = 100$